



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/733,037	12/11/2003	Keith J. Purcell	RSW920030159US1	8444

23307 7590 08/21/2006

SYNNESTVEDT & LECHNER, LLP
2600 ARAMARK TOWER
1101 MARKET STREET
PHILADELPHIA, PA 191072950

EXAMINER

CHEN, QING

ART UNIT	PAPER NUMBER
----------	--------------

2194

DATE MAILED: 08/21/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

2

Office Action Summary	Application No.	Applicant(s)	
	10/733,037	PURCELL, KEITH J.	
	Examiner	Art Unit	
	Qing Chen	2191	

– The MAILING DATE of this communication appears on the cover sheet with the correspondence address –
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 11 December 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-27 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-27 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 11 December 2003 is/are: a) ☐ accepted or b) ☒ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>20031211</u> . | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This is the initial Office action based on the application filed on December 11, 2003.

Claims 1-27 are currently pending and have been considered below.

Information Disclosure Statement

2. The Office acknowledges receipt of the Information Disclosure Statement filed on December 11, 2003. It has been placed in the application file and the information referred to therein has been considered by the Examiner.

Drawings

3. The drawings are objected to because Figure 1 contains mislabeled drawing elements. Elements 16a-16f should be labeled as "XSL style sheet." Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application.

The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference character(s) not mentioned in the description:

- Element 29 in Figure 2.

Corrected drawing sheets in compliance with 37 CFR 1.121(d), or amendment to the specification to add the reference character(s) in the description in compliance with 37 CFR 1.121(b) are required in reply to the Office action to avoid abandonment of the application.

Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as “amended.” If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either “Replacement Sheet” or “New Sheet” pursuant to 37 CFR 1.121(d). If the changes are not accepted by the Examiner, the Applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Specification

4. The disclosure is objected to because of the following informalities:
 - The specification contains the following typographical errors:
 - The sentence “In the exemplary implementation, the programming models comprise ...” contains numerous typographical errors in page 4, paragraph [0012]. The sentence should be changed to “In the exemplary implementation, the programming models comprise XSL style sheets (16a-16f) and XML templates (18a-18f).”
 - The acronym XSL should be added in between the words “a” and “style sheet” in page 5, paragraph [0015].

Art Unit: 2191

- The word “and” in front of the phrase “XML/XSL transform is applied” should be deleted in page 6, paragraph [0016].
- The trademark “Java Server” should be written as “JavaServer” in page 6, paragraph [0016].
- A right angle bracket (>) should be added at the end of the closing “object_context” tag in page 7, paragraph [0018], and line 6.
- All occurrences of the term “gif” in the specification should be capitalized.
- All occurrences of the term “eXtensible Style Language” in the specification should be changed to “eXtensible Stylesheet Language.”
- All occurrences of the term “stylesheet” in the specification should be changed to “style sheet.” Although the two conventions are used interchangeably, applicant is advised to make the correction in order to keep the terminology consistent throughout the specification and claims.

Appropriate correction is required.

5. The use of trademarks, such as JAVA, J2EE, IBM, and JAVASERVER, has been noted in this application. Trademarks should be capitalized wherever they appear (capitalize each letter OR accompany each trademark with an appropriate designation symbol, e.g., TM or ®) and be accompanied by the generic terminology (use trademarks as adjectives modifying a descriptive noun, e.g., “the JAVA programming language”).

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner, which might adversely affect their validity as trademarks.

Claim Objections

6. **Claims 1, 4-12, 16-19, 21, 22, and 24-26** are objected to because of the following informalities:

- **Claim 1** recites the limitation “said object” in the last limitation. The Examiner subsequently interprets this limitation as reading “said output object” for the purpose of providing it with proper explicit antecedent basis.
- **Claims 1, 8, 9, 12, 17-19, 25, and 26** contain a typographical error: the word “and” should be added in between the semicolon (;) and the last limitation.
- **Claims 4-11** contain a typographical error: the article used to designate the method should be changed from “A” to “The.”
- **Claims 7, 16, and 24** contain a typographical error: the term “eXtensible Style Language” in the specification should be changed to “eXtensible Stylesheet Language.”
- **Claim 11** contains a typographical error: the letter “j” in “java” should be capitalized.
- **Claim 12** contains a typographical error: the word “have” should be changed to “having” in the second limitation.
- **Claims 17 and 18** contain a typographical error: the category of invention “computer programming product” should be changed to “computer program product” as indicated in the parent claim.

Art Unit: 2191

- **Claims 19, 21, 25, and 26** recite the limitation “said description.” The Examiner subsequently interprets this limitation as reading “said application description” for the purpose of providing it with proper explicit antecedent basis.
- **Claim 22** contains a typographical error: claim 22 should depend from claim 19, not claim 18, since claim 19 is the independent claim of the system.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

7. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

8. **Claim 12** is rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim contains subject matter, which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

The claimed invention omits the critical step of generating an output object disclosed to be essential to the invention. The step of returning the object cannot occur without a prior step of generating the object or an indication of generating the object as claimed in claim 1. A claim which omits matter disclosed to be essential to the invention as described in the specification or in other statements of record may be rejected under 35 U.S.C. 112, first paragraph, as not enabling. *In re Mayhew*, 527 F.2d 1229, 188 USPQ 356 (CCPA 1976). See also MPEP §

Art Unit: 2191

2164.08(c). Such essential matter may include missing elements, steps or necessary structural cooperative relationships of elements described by the applicant as necessary to practice the invention.

9. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

10. **Claims 8-12, 15, 16, and 27** are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claims 8 and 9 recite the limitation “said object template.” There is insufficient antecedent basis for this limitation in the claims. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “said coding module” for the purpose of further examination.

Claims 10, 11, and 27 contain the trademarks or trade names IBM, WEBSHERE, and JAVA. When a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of the 35 U.S.C. 112, second paragraph. *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, the use of a trademark or trade name in a claim to identify

Art Unit: 2191

or describe a material or product would not only render a claim indefinite, but would also constitute an improper use of the trademark or trade name.

Claim 12 recites the limitation “returning object” in the last limitation. There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “returning an object” for the purpose of further examination. Since the claim contains no earlier recitation or limitation of an object—except an object template—and it is unclear as to what element the limitation was making reference to, the Examiner also subsequently interprets this limitation as reading with a prior step of generating an object by applying the description to the object template.

Claims 15 and 16 recite the limitation “said coding module.” There is insufficient antecedent basis for this limitation in the claims. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “said object template” for the purpose of further examination.

11. **Claim 12** is rejected under 35 U.S.C. 112, second paragraph, as being incomplete for omitting an essential step, such omission amounting to a gap between the steps. See MPEP § 2172.01.

The omitted step is: generating an output object prior to returning it. The omitted step is considered to be critical to the claimed invention since the step is necessary and must occur in the method for the claimed invention to function as intended as described in the specification.

Art Unit: 2191

The step of returning the object cannot occur without a prior step of generating the object or an indication of generating the object as claimed in claim 1.

Claim Rejections - 35 USC § 101

12. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

13. **Claims 12-18** are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 12-18 are directed to functional descriptive material *per se*, since the computer program products are not claimed as being embodied in a computer-readable medium, and hence non-statutory. The claims constitute computer programs representing computer listings *per se*. Such descriptions or expressions of the programs are not physical “things.” They are neither computer components nor statutory processes, as they are not “acts” being performed. Such claimed computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer, which permit the computer program’s functionality to be realized. In contrast, a claimed computer-readable medium encoded with a computer program is a computer element, which defines structural and functional interrelationships between the computer program and the rest of the computer, that permits the computer program’s functionality to be realized, and is thus statutory. See *Lowry*, 32 F.3d at 1583-84, 32 USPQ2d at 1035.

Claim Rejections - 35 USC § 102

14. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

15. **Claims 1-9 and 11-26** are rejected under 35 U.S.C. 102(e) as being anticipated by

Hejlsberg et al. (US 2004/0088688).

As per **Claim 1**, Hejlsberg et al. disclose a method for automatically generating computer program code comprising the steps of:

- A. Generating a description of an application (*see Figure 2, Element 200; and Paragraphs [0006] and [0034]*);
- B. Providing said description to a web service (*see Paragraphs [0017], [0032], and [0087]*);
- C. Parsing said description by said web service (*see Paragraphs [0035] and [0058]*);
- D. Locating a suitable coding module on a node contained within a computational grid (*see Paragraphs [0036]-[0038]*);
- E. Supplying said description to said node (*see Paragraphs [0041] and [0042]*);

Art Unit: 2191

F. Applying said description to said coding module to generate an output object (*see Figure 2, Element 220; and Paragraph [0043]*); and

G. Returning said output object (*see Paragraphs [0016] and [0060]*).

As per **Claim 2**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 1 above**, and further disclose that the suitable coding module comprises a plurality of coding modules (*see Paragraphs [0039] and [0041]*).

As per **Claim 3**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 2 above**, and further disclose that the plurality of coding modules is located on a plurality of nodes within a computational grid (*see Paragraph [0031]*).

As per **Claim 4**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 1 above**, and further disclose that the description is generated using Object Meta Language (OML) (*see Paragraph [0047]*).

As per **Claim 5**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 4 above**, and further disclose that the OML is an eXtensible Markup Language (XML) dialect (*see Paragraphs [0006] and [0047]*).

As per **Claim 6**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 1 above**, and further disclose that the coding module is an XML template (*see Paragraphs [0017] and [0032]*).

As per **Claim 7**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 1 above**, and further disclose that the coding module is an eXtensible Stylesheet Language (XSL) style sheet (*see Paragraph [0017]*).

As per **Claim 8**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 7 above**, and further disclose that the step of applying said description to said coding module comprises the steps of:

- A. Parsing said description to locate at least one variable (*see Paragraph [0048]*); and
- B. Substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is the result of an XML/XSL transform (*see Paragraphs [0048] and [0050]*).

As per **Claim 9**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 6 above**, and further disclose that the step of applying said description to said coding module comprises the steps of:

- A. Parsing said description to locate at least one variable (*see Paragraph [0048]*); and

B. Substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is stored in said XML template (*see Paragraphs [0048] and [0050]*).

As per **Claim 11**, Hejlsberg et al. disclose a method for automatically generating computer program code **as in Claim 1 above**, and further disclose that the output object is a Java™ file (*see Paragraph [0051]*).

As per **Claim 12**, Hejlsberg et al. disclose a computer program product for automatically generating computer program code, comprising computer executable instructions for:

- A. Generating a description of an application (*see Figure 2, Element 200; and Paragraphs [0006] and [0034]*);
- B. Providing said description to a web service (*see Paragraphs [0017], [0032], and [0087]*);
- C. Parsing said description by said web service (*see Paragraphs [0035] and [0058]*);
- D. Locating a suitable object template on a node contained within a computational grid (*see Paragraphs [0017], [0032], [0037], and [0038]*);
- E. Supplying said description to said node (*see Paragraphs [0041] and [0042]*);
- F. Applying said description to said object template (*see Figure 2, Element 220; and Paragraphs [0017], [0032], and [0043]*); and
- G. Returning an object (*see Paragraphs [0016] and [0060]*).

As per **Claim 13**, Hejlsberg et al. disclose a computer program product for automatically generating computer program code **as in Claim 12 above**, and further disclose that the description comprises Object Meta Language (OML) (*see Paragraph [0047]*).

As per **Claim 14**, Hejlsberg et al. disclose a computer program product for automatically generating computer program code **as in Claim 12 above**, and further disclose that the OML is an eXtensible Markup Language (XML) dialect (*see Paragraphs [0006] and [0047]*).

As per **Claim 15**, Hejlsberg et al. disclose a computer program product for automatically generating computer program code **as in Claim 12 above**, and further disclose that the object template is an XML template (*see Paragraphs [0017] and [0032]*).

As per **Claim 16**, Hejlsberg et al. disclose a computer program product for automatically generating computer program code **as in Claim 12 above**, and further disclose that the object template is an eXtensible Stylesheet Language (XSL) style sheet (*see Paragraph [0017]*).

As per **Claim 17**, Hejlsberg et al. disclose a computer program product for automatically generating computer program code **as in Claim 15 above**, and further disclose that the computer readable instructions for applying said description to said object template comprise instructions for:

A. Parsing said description to locate at least one variable (*see Paragraph [0048]*); and

B. Substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is the result of an XML/XSL transform (*see Paragraphs [0048] and [0050]*).

As per **Claim 18**, Hejlsberg et al. disclose a computer program product for automatically generating computer program code **as in Claim 15 above**, and further disclose that the computer readable instructions for applying said description to said object template further comprise instructions for:

- A. Parsing said description to locate at least one variable (*see Paragraph [0048]*); and
- B. Substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is stored in said XML template (*see Paragraphs [0048] and [0050]*).

As per **Claim 19**, Hejlsberg et al. disclose a system for automatically generating computer program code comprising:

- A. An input terminal for inputting an application description (*see Figure 1, Element 110; and Paragraph [0022]*);
- B. A computational grid having at least one node (*see Paragraphs [0017] and [0032]*);
- C. A web service for supplying said application description to said node (*see Paragraphs [0017] and [0032]*); and
- D. A coding module residing on said node, wherein said coding module generates an object from said description (*see Paragraph [0043]*).

As per **Claim 20**, Hejlsberg et al. disclose a system for automatically generating computer program code **as in Claim 19 above**, and further disclose that the coding module comprises a plurality of coding modules (*see Paragraphs [0039] and [0041]*).

As per **Claim 21**, Hejlsberg et al. disclose a system for automatically generating computer program code **as in Claim 19 above**, and further disclose that the application description is generated using Object Meta Language (OML) (*see Paragraph [0047]*).

As per **Claim 22**, Hejlsberg et al. disclose a system for automatically generating computer program code **as in Claim 19 above**, and further disclose that the OML is an eXtensible Markup Language (XML) dialect (*see Paragraphs [0006] and [0047]*).

As per **Claim 23**, Hejlsberg et al. disclose a system for automatically generating computer program code **as in Claim 19 above**, and further disclose that the coding module is an XML template (*see Paragraphs [0017] and [0032]*).

As per **Claim 24**, Hejlsberg et al. disclose a system for automatically generating computer program code **as in Claim 19 above**, and further disclose that the coding module is an eXtensible Stylesheet Language (XSL) style sheet (*see Paragraph [0017]*).

As per **Claim 25**, Hejlsberg et al. disclose a system for automatically generating computer program code **as in Claim 24 above**, and further disclose that the coding module for generating an object from said application description includes computer code for:

- A. Parsing said description to locate at least one variable (*see Paragraph [0048]*); and
- B. Substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is the result of an XML/XSL transform (*see Paragraphs [0048] and [0050]*).

As per **Claim 26**, Hejlsberg et al. disclose a system for automatically generating computer program code **as in Claim 23 above**, and further disclose that the coding module for generating an object from said application description includes computer code for:

- A. Parsing said description to locate at least one variable (*see Paragraph [0048]*); and
- B. Substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is stored in said XML template (*see Paragraphs [0048] and [0050]*).

Claim Rejections - 35 USC § 103

16. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

17. **Claims 10 and 27** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Hejlsberg et al.** (US 2004/0088688) in view of **Burke et al.** (US 6,789,252).

As per **Claim 10**, **Hejlsberg et al.** disclose a method for automatically generating computer program code **as in Claim 1 above**. However, **Hejlsberg et al.** does not explicitly disclose that the web service is IBM® WebSphere®.

In the same field of endeavor, **Burke et al.** disclose a method and apparatus for creating and applying dynamically defined business objects used in computer systems, for using business objects to configure business software applications, and for facilitating the automated sharing of business information across a business enterprise or with other business enterprises or customers. In the system of **Burke et al.**, the web server computer utilizes IBM® WebSphere® (*see Column 33, Lines 23-36*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to utilize IBM® WebSphere® as the web service application in the system of **Hejlsberg et al.** because **Hejlsberg et al.** already disclose utilizing a web service in his system (*see Paragraphs [0017], [0032], and [0087]*) and IBM® WebSphere® has been used widely as

Art Unit: 2191

a web service application to operate and integrate applications across multiple computing platforms using web technologies as supported by the system of Burke et al. One would have been motivated to utilize IBM® WebSphere® as the web service application because it supports enterprise Java™ open standards.

As per **Claim 27**, Hejlsberg et al. disclose a system for automatically generating computer program code as in **Claim 19 above**. However, Hejlsberg et al. does not explicitly disclose that the web service is IBM® WebSphere®.

In the same field of endeavor, Burke et al. disclose a method and apparatus for creating and applying dynamically defined business objects used in computer systems, for using business objects to configure business software applications, and for facilitating the automated sharing of business information across a business enterprise or with other business enterprises or customers. In the system of Burke et al., the web server computer utilizes IBM® WebSphere® (*see Column 33, Lines 23-36*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to utilize IBM® WebSphere® as the web service application in the system of Hejlsberg et al. because Hejlsberg et al. already disclose utilizing a web service in his system (*see Paragraphs [0017], [0032], and [0087]*) and IBM® WebSphere® has been used widely as a web service application to operate and integrate applications across multiple computing platforms using web technologies as supported by the system of Burke et al. One would have been motivated to utilize IBM® WebSphere® as the web service application because it supports enterprise Java™ open standards.

Conclusion

18. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

A. **Lindsey** (US 5,675,801) discloses a system and method in which a user prototypes desired programming functions utilizing an object-oriented user interface to create an object-oriented design of the desired programming function. The object-oriented design is matched to source code templates for the target language, the templates are parsed by a generator engine, and object-oriented code is executed to produce source code files in the target language.

B. **Garloff et al.** (US 5,699,310) disclose a computer system wherein object-oriented management techniques are used with a new means for generating code to provide for the automatic generation of source code.

C. **Sadiq et al.** (US 5,978,581) disclose a system and method for object-oriented code generation based on an object model.

D. **Lau** (US 6,598,219) discloses a mechanism for a task-oriented data model for a development tool in an object-oriented programming system.

E. **Kumar** (US 7,017,148) discloses a method and apparatus for UPnP device code generation using XML.

F. **Uehara et al.** (US 2003/0212956) disclose a program generating apparatus, a program generating method and a program generating program (also known as program generator or simply as generator) for generating a program by combining, linking, or concatenating structural components of the program in a desired form.

G. **Curry et al.** (US 2003/0233631) disclose a method for rapid design, development, deployment and support of web applications based on web services with minimum customized programming, maximized reuse of software components and compliance with standard development frameworks.

H. **Al-Azzawe et al.** (US 2004/0049736) disclose a system and method for automatically generating code for converting data from stored procedures to an XML format.

I. **Kayam et al.** (US 2004/0268296) disclose a system and associated methods for software assembly.

J. **Arcaro et al.** (US 2005/0177817) disclose a software for generating computer code of at least one part of a computer application from a description of the part of the computer application.

K. **Harrop et al.** (US 2006/0036612) disclose a document assembly system, and methods executed by components of the system.

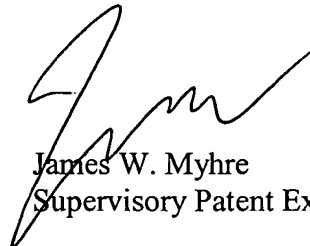
Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, James W. Myhre, can be reached on 571-270-1065. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2191

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

QC / *QC*
August 14, 2006



James W. Myhre
Supervisory Patent Examiner